

Comment aborder l'élevage de précision dans l'enseignement agricole ? : approche à partir d'exemples

Thibault Maillot, L'Institut Agro Dijon

CHAPITRE I - Identification électronique, sas de tri/contention

Matériel pédagogique

Corrigés

Objectifs et matériel utilisé

Durant cette séance de TP, vous vous servirez d'outils (capteurs, puce RFID, ...) afin de comprendre leurs principes de fonctionnement et leurs utilisations en élevage de précision.

Les objectifs de la séance sont :

1. Étudier et utiliser des méthodes d'identification d'un animal.
2. Mettre en place une solution (simplifiée) de tri des animaux.

ATTENTION : Une préparation préalable des tags RFID est nécessaire. Pour cela, il faut utiliser le programme emulateSAS_initTag.ino contenu dans le matériel pédagogique du chapitre.

Table des matières

1. Test de la distance de lecture d'un lecteur de puces RFID	2
2. Mise en place d'un sas de tri	5
Annexes	
A. Rapporteur à imprimer en A3	7
Références	8

1. Test de la distance de lecture d'un lecteur de puces RFID

Objectifs et matériel utilisé :

L'objectif de ce TP est d'utiliser et de tester, via une carte Arduino, un lecteur de puces RFID.

Les objectifs de la séance sont :

1. Utiliser une carte Arduino pour gérer un lecteur de puces RFID.
2. Utiliser le système pour tester la capacité de lecture du contrôleur RFID.

Afin de compléter ces objectifs, vous utiliserez le matériel suivant :

- Une carte Arduino et son câble USB.
- Un contrôleur NFC/RFID adapté à l'Arduino (PN532 NFC/RFID Controller Shield de chez Adafruit [1] ou équivalent).
- Un ou plusieurs tag RFID 13.56 MHz (compatible avec le lecteur RFID [1]).
- Un ordinateur avec le logiciel Arduino installé.

1.1. Utilisation du lecteur RFID avec l'Arduino

Le projet sur lequel vous allez travailler est contenu dans le dossier *USERFID*. Le fichier principal du projet est nommé *USERFID.INO*. C'est ce fichier que l'on vous demandera de modifier.

Les deux fichiers *LIB_TP1.H* et *LIB_TP1.CPP* définissent des fonctionnalités que vous allez devoir utiliser, dans le fichier principal.

QUESTIONS

Répondez aux questions suivantes :

1. Lisez, comprenez et commentez le fichier *lib_TP1.h*.

2. Quelle est la fonction qui permet l'**initialisation** des composants ?

`bool initialisation_RFID(void) ;`

3. À quoi sert la structure *TagRfidStruct* ?

La structure contient le format de données comme elles sont stockées dans les tags RFID.

4. Comment pouvez-vous lire un badge RFID ?

`bool lireTag(TagRfidStruct& tagRfidStruct) ;`

5. À quoi sert le mot `#define` ?

Il sert à déclarer des variables constantes.

6. Dans le fichier *useRFID.ino* :

a. Modifiez la fonction *setup()* pour **initialiser** le module RFID.

b. Modifiez la fonction *loop()* pour **lire** une puce RFID (si elle est présente) et afficher son contenu dans un moniteur série.

c. Modifiez la fonction *loop()* pour **allumer** une LED **si** un animal est détecté lors de la **lecture** du tag RFID.

Voir *useRFID_corr.ino*

1.2. Zone de détection du capteur

On souhaite connaître la zone de détection du lecteur RFID, c'est-à-dire, pour une position fixée du lecteur RFID, l'ensemble des positions d'un tag RFID pour lesquelles sa lecture est possible.

Pour la suite, vous allez utiliser le système développé dans la partie précédente, ainsi que le rapporteur de l'Annexe A.

QUESTIONS

Répondez aux questions suivantes :

1. Positionnez le lecteur RFID de telle sorte que l'antenne corresponde au centre du rapporteur, comme montré sur la Figure 1.
2. Pour chaque dizaine de degrés :
 - a. Positionnez un tag RFID sur la ligne d'angle correspondante, à l'endroit le plus éloigné du lecteur RFID.
 - b. Tant que le lecteur ne détecte pas le tag RFID (la LED reste éteinte), avancez le tag de centimètre en centimètre.
 - c. Lorsque le lecteur RFID détecte le tag RFID, marquez l'endroit correspondant d'un point.
3. Une fois toutes les positions de détection marquées, tracez la zone de détection en les reliant. Que constatez-vous ?

► Voir Figure 1-corr page suivante.

La zone de détection forme une parabole avec un maximum de distance de lecture de 6 cm, environ.

4. Pourquoi ne pas avoir fait des mesures « derrière » le capteur ? Vous pouvez faire quelques essais pour justifier votre réponse.

La prise de mesure « derrière » le capteur est tout à fait possible. L'antenne étant symétrique, le résultat de la détection aurait été quasi identique à ce qui a été observé à la question 3.

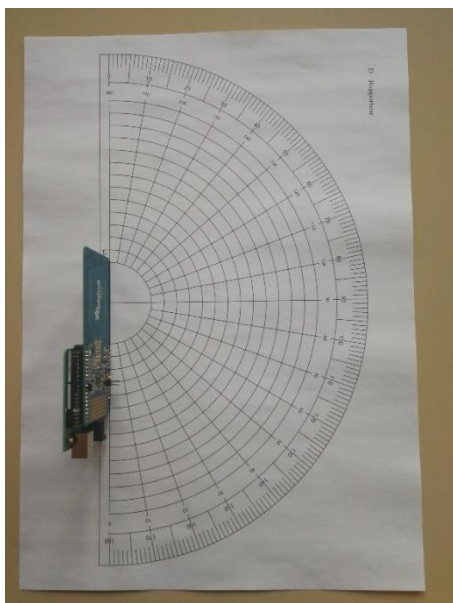


Figure 1 : Positionnement du lecteur RFID

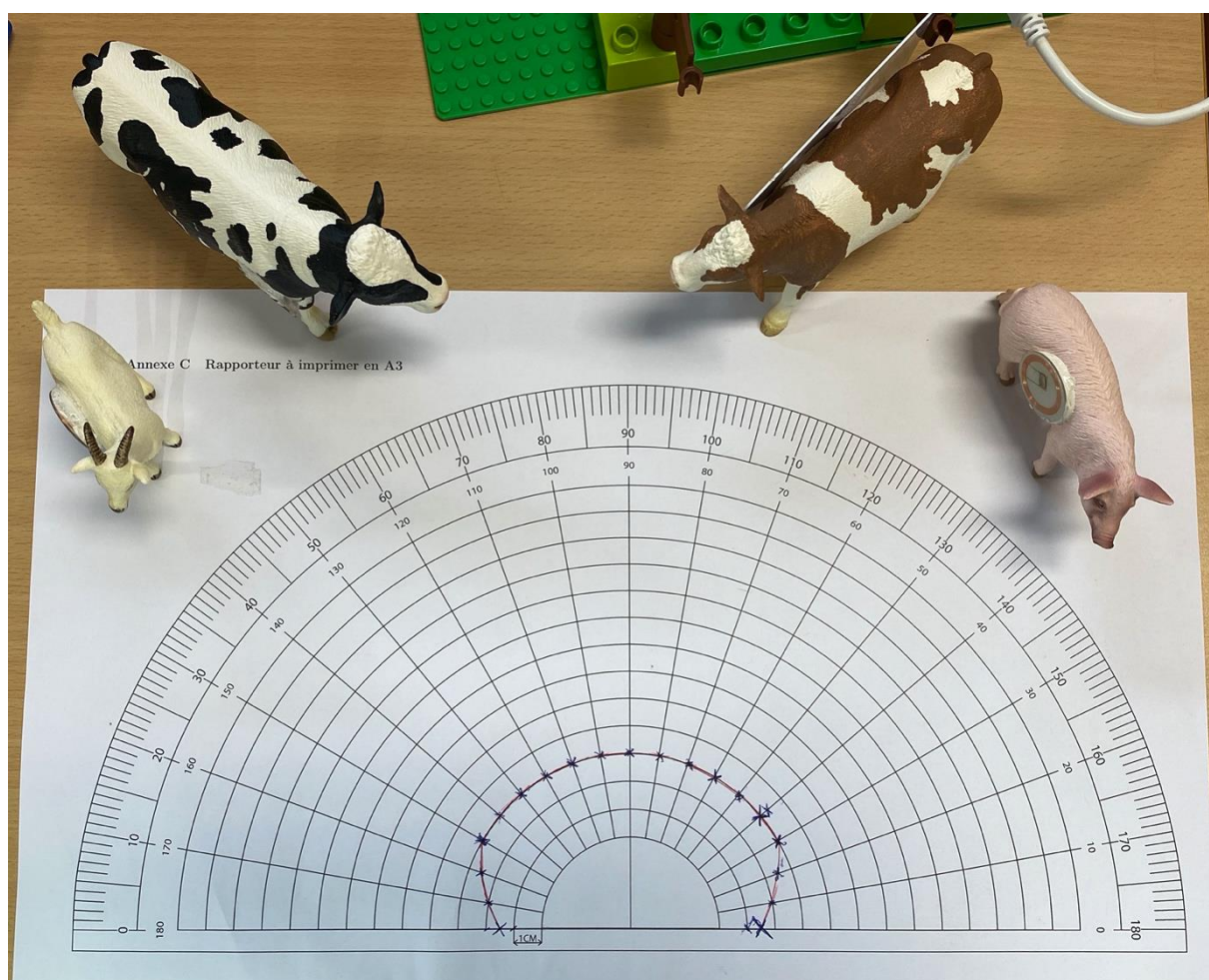


Figure 1-corr
 (A. Bras, U. Dubroeuq, L. Duffo, A. Maquet, TP1 : Utilisation d'un module RFID,
 Compte-rendu du TP1 du module Agriculture de précision, M1 GETIA, 2020)

2. Mise en place d'un sas de tri

Une maquette est à votre disposition afin de simuler un sas de tri utilisant l'identification par puce RFID. L'objectif est de répondre à la problématique suivante :

« Un randonneur a oublié de fermer une barrière d'un pré. Deux lots d'animaux se sont mélangés (des vaches Holstein et Simmental française). Nous souhaitons mettre en place une solution automatisée afin de trier les lots d'animaux et les reconduire dans leurs parcs respectifs. »

Afin de simuler le problème, le matériel suivant vous est fourni :

- Une prairie en LEGO®.
- Des vaches des deux races munies de tags RFID préconfigurés.
- Un contrôleur NFC.
- Un servomoteur relié à une porte.
- Un Arduino UNO pour lire les tags RFID et commander la porte.

QUESTIONS

Répondez aux questions suivantes :

1. Le projet à utiliser est contenu dans le dossier EMULATESAS. Le fichier principal du projet est nommé EMULATESAS.INO. Le fichier LIB_TP1.H du projet est identique à celui de la section 1.1.

2. En utilisant les fonctions du fichier LIB_TP1.H, comment pouvez-vous faire **bouger** le servomoteur ?
`void bougeServo(int angle) ;`

3. Dans le fichier EMULATESAS.INO :

- Modifiez la fonction setup() pour **initialiser** le module RFID et le servomoteur.
- Modifiez la fonction loop() pour lire une puce RFID (**si** elle est présente) et afficher son contenu dans un moniteur série.
- Modifiez la fonction loop() pour faire **bouger** le servomoteur en fonction de **l'espèce** de l'animal détecté lors de la lecture du tag RFID (la Remarque 1 peut vous aider).
- (*facultatif*) Modifiez la fonction loop() pour **mettre à jour** le nombre de **passages** du tag RFID devant le contrôleur NFC.

La correction est disponible dans `emulateSAS_corr`.

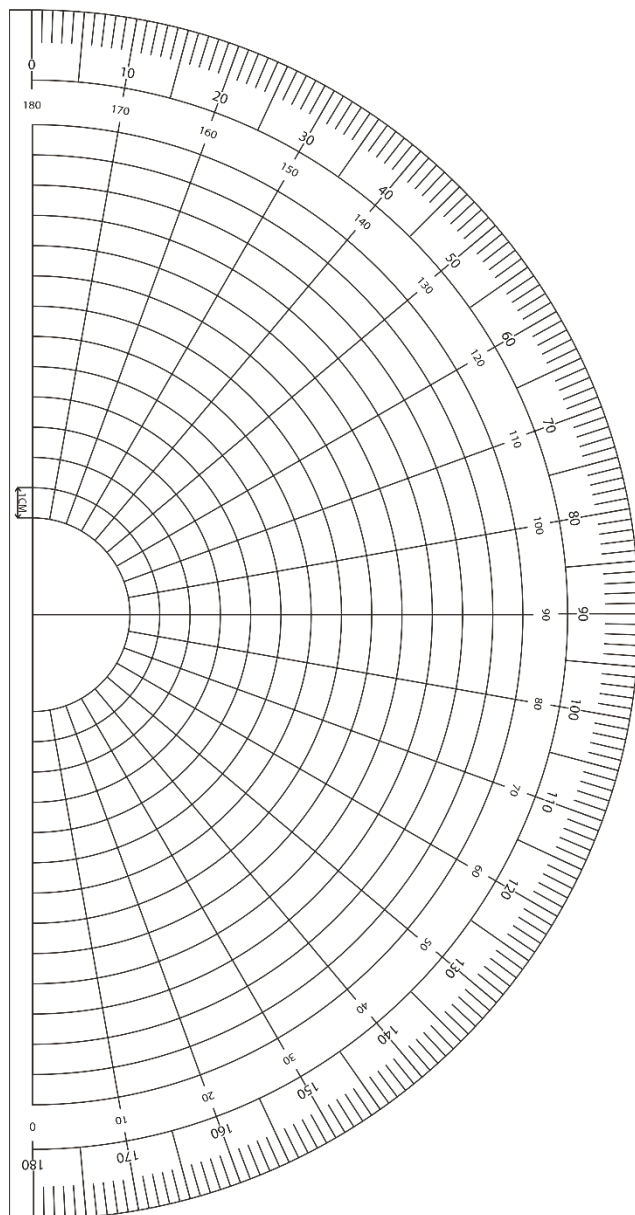
REMARQUE 1 (Comparaison de chaîne de caractères)

La fonction strcmp permet de comparer deux chaînes de caractères.

La syntaxe est la suivante : `res = strcmp(chaine1, chaine2)` ; res vaut 0 si les deux chaînes sont identiques.

ANNEXES

A - Rapporteur à imprimer en A3



RÉFÉRENCES

- [1] Adafruit, « Adafruit PN532 RFID/NFC Breakout and Shield », Consulté en 2017.
<https://learn.adafruit.com/adafruit-pn532-rfid-nfc>